

# **LAPORAN PRAKTIKUM 5**

**Mata Kuliah**

**“Pemograman Berorientasi Obyek”**



**Disusun Oleh :**

**Dian Nur Safitri  
21091397044\_2021B**

**PROGRAM STUDI D4 MANAJEMEN INFORMATIKA**

**FAKULTAS VOKASI**

**UNIVERSITAS NEGERI SURABAYA**

1. Suatu program terdiri dari class Pegawai sebagai parent class, class Manajer dan class Kurir sebagai subclass. Buatlah suatu program yang menerapkan konsep polymorphic argument (Polymorphic arguments adalah tipe suatu parameter yang menerima suatu nilai yang bertipe subclass-nya).

## Source Code Php

```
Praktikum 5 No.1.php - Visual Studio Code

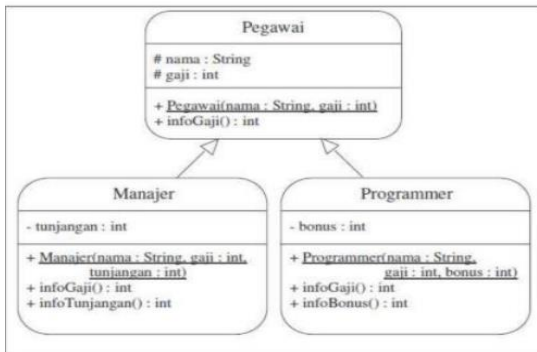
C:\Users\DIAN NUR SARITRI\Downloads\PBO UTS> Praktikum 5 No.1.php
1 <!-- Nama : Dian Nur Safitri -->
2 <!-- NIM : 21091397044 -->
3 <!-- Kelas: 2021B / D4 Manajemen Informatika -->
4 <!-- Soal NO 1 Praktikum 5 -->
5
6 <!DOCTYPE html>
7 <html lang="id">
8 <head>
9     <meta charset="UTF-8">
10    <meta http-equiv="X-UA-Compatible" content="IE=edge">
11    <meta name="viewport" content="width=device-width, initial-scale=1.0">
12    <title>Praktikum 5</title>
13 </head>
14 <body style="align-items: center;">
15     <div class="container">
16         <h2>Soal 1</h2>
17     </div>
18     <?php
19     class Pegawai
20     {
21         public $nama;
22         public function __construct($nama)
23         {
24             $this->nama = $nama;
25         }
26         public function getNama()
27         {
28             return $this->nama;
29         }
30     }
31     class Manajer extends Pegawai
32     {
33         public $tunjangan;
34         public function __construct($nama, $tunjangan)
35         {
36             parent::__construct($nama);
37             $this->tunjangan = $tunjangan;
38         }
39         public function getTunjangan()
40         {
41             return $this->tunjangan;
42         }
43     }
44     class Kurir extends Pegawai
45     {
46         public $gaji;
47         public function __construct($nama, $gaji)
48         {
49             parent::__construct($nama);
50             $this->gaji = $gaji;
51         }
52         public function getGaji()
53         {
54             return $this->gaji;
55         }
56     }
57     class Soal1
58     {
59         public static
60         function Proses($peg)
61         {
62         }
```

```
64 {
65     if ($peg instanceof Manajer)
66     {
67         $man = $peg;
68         echo "<br>Nama manajer: ".$man->nama, "\n";
69         echo "<br>Tunjangan: Rp. ".strval($man->tunjangan), "\n";
70     }
71     else if ($peg instanceof Kurir)
72     {
73         $kur = $peg;
74         echo "<br>Nama kurir: ".$kur->nama, "\n";
75         echo "<br>Gaji: Rp. ".strval($kur->gaji), "\n";
76     }
77 }
78
79 public static
80 function main($args)
81 {
82     echo "Dian Nur Safitri- 21091397044", "\n";
83     echo "<br>", "<br>";
84     $peg1 = new Manajer("safitri", 9000000);
85     Soal1::Proses($peg1);
86     echo "<br>", "<br>";
87     $peg2 = new Kurir("Ica ", 7500000);
88     Soal1::Proses($peg2);
89 }
90
91 Soal1::main(array());
92 }>
93
94 </div>
95 </body>
96 </html>
```

## Penjelasan :

- Pada kasus soal diatas, yaitu implementasi polymorphic argument dengan berada di class info dimana method info dibuat static supaya pemanggilnya tidak memerlukan diinisiasi. Sehingga dapat langsung dimasukkan menjadi parameter pada method info di class info.

2. Buat program berdasarkan UML berikut



## Source Code Php

```
Praktikum 5 No.2.php X
C:\Users\DIAN NUR SAFITRI\Downloads\PBO UTS\Praktikum 5 No.2.php
1 <!-- Nama : Dian Nur Safitri-->
2 <!-- NIM : 21091397044 -->
3 <!-- Kelas: 2021B / D4 Manajemen Informatika -->
4 <!-- Soal NO 2 Praktikum 5 -->
5
6 <!DOCTYPE html>
7 <html lang="id">
8 <head>
9     <meta charset="UTF-8">
10    <meta http-equiv="X-UA-Compatible" content="IE=edge">
11    <meta name="viewport" content="width=device-width, initial-scale=1.0">
12    <title>Praktikum 5</title>
13 </head>
14 <body>
15     <div class="container">
16         <h2>Soal 2</h2>
17     </div>
18     <?php
19     class Pegawai
20     {
21         public $nama;
22         public $gaji;
23         public function __construct($nama, $gaji)
24         {
25             $this->nama = $nama;
26             $this->gaji = $gaji;
27         }
28         public function infoGaji()
29         {
30             return $this->gaji;
31         }
32     }
33
34     class Manajer extends Pegawai
35     {
36         private $tunjangan;
37         public function __construct($nama, $gaji, $tunjangan)
38         {
39             parent::__construct($nama, $gaji);
40             $this->tunjangan = $tunjangan;
41         }
42         public function infoGaji()
43         {
44             return $this->gaji;
45         }
46         public function infoTunjangan()
47         {
48             return $this->tunjangan;
49         }
50     }
51
52     class Programmer extends Pegawai
53     {
54         private $bonus;
55         public function __construct($nama, $gaji, $bonus)
56         {
57             parent::__construct($nama, $gaji);
58             $this->bonus = $bonus;
59         }
60         public function infoGaji()
61         {
62             return $this->gaji;
63         }
64         public function infoBonus()
65         {
66             return $this->bonus;
67         }
68     }
69 }
```

```
65     }
66 }
67 class Bayaran
68 {
69     public function hitungBayaran($peg)
70     {
71         $uang = $peg->infoGaji();
72
73         return $uang;
74     }
75     public static function main($args)
76     {
77         echo "Dian Nur Safitri- 21091397044";
78         echo "<br>";
79         $man = new Manajer["Safitri", 6000000, 800000];
80         echo "<br>", "<br>";
81         $prog = new Programmer("Ica", 10000000, 900000);
82         echo "<br>", "<br>";
83         $hr = new Bayaran();
84         echo "<br> Gaji untuk Manajer Bernama ". $man->nama. " : Rp. ".strval($hr->hitungBayaran($man)) , "<br>";
85         echo "<br>", "<br>";
86         echo "<br>Gaji untuk Programmer Bernama ". $prog->nama. " : Rp. ".strval($hr->hitungBayaran($prog)) , "<br>";
87     }
88 }
89
90 Bayaran::main(array());
91 }>
92 <!-- END PHP -->
93 </div>
94 </body>
95 </html>
```

### Penjelasan :

- Kasus soal diatas, yaitu Program penerapan inheritance dengan konsep overriding ini, pemanggilan construct di masing-masing class turunannya hanya akan menginisiasi property yang dimiliki dengan visibilitas private dan property lain yang diturunkan akan langsung diinisiasi dengan construct dari parentnya